基于匹配策略融合的低误差快速图像拼接算法 *

杨明东，石　英†，华逸伦，朱剑怀

(武汉理工大学 自动化学院，武汉 430070)

摘　要：针对目前多数拼接算法正确匹配率低、耗时及误差大的问题，提出了基于匹配策略融合的改进图像拼接算法。该算法首先仅在图像的重合区域提取 SIFT 特征点，并计算 SURF 特征描述符；其次，融合改进的最近邻比次近邻、双向交叉检查及匹配差值的阈值化 3 种匹配策略，结合坐标约束及 RANSAC 算法完成特征匹配；最后，提出利用配准参数计算任意图像到参考图像坐标空间的投影变换模型的方法，并利用多线程技术将所有图像投影至参考坐标系，经过亮度校正、加权融合后合成全景图。实验结果表明，提出的拼接算法正确匹配率提高了 10~20%，拼接总耗时约为传统逐帧扩大式拼接算法的 1/3，且累计误差大大降低，拼接图像畸变小。

关键词：匹配策略融合；快速图像拼接；累计误差；投影变换模型；多线程

中图分类号：TP391.41　　doi: 10.3969/j.issn.1001-3695.2017.11.0772

Low-error and fast image-stitching algorithm based on merging match strategies

Yang Mingdong, Shi Ying, Hua Yilun, Zhu Jianhuai

(*School of Automation*, *Wuhan university of technology*, *Wuhan Hubei* 430070, *China*)

Abstract: To deal with low correct matching, high time-consuming and high-error in most stitching algorithms, this paper developed an improved image stitching algorithm based on merging match strategies. First, the algorithm extracted SIFT feature points only from the overlap area of pair-image, and computed the SURF descriptors; Second, the algorithm merged three match strategies including ratio of the nearest neighbor and the 2nd nearest neighbor distances, cross-check of two directions and pair-matches' difference threshold method, combined coordinate constraint with RANSAC algorithm to complete the feature match; Finally, the algorithm used registration data to calculate the projection model which could transform any image in the sequences to reference image coordinate space, then used multithread technology to accelerate. After the brightness correction, the algorithm used the weighted fusion method to composite panorama images. The experimental results show that the proposed algorithm obtains a 10~20% correct rate increase in feature match with about 1/3 time-consuming of the traditional frame-to-frame expand stitching algorithm, great decrease in accumulated error, and little image distortion.

Key words: merged match strategies; fast image stitching; accumulated error; projection transformation model; multithread

0　引言

图像拼接技术将同一场景的多幅序列图像经过配准、融合后合成宽视野、高分辨率的全景图，近年来，广泛应用于三维重建、视频监控、遥感影像及灾害防控等领域[1,2]。图像配准是拼接算法的核心步骤，直接决定了拼接算法的实时性和累计误差[3]，为了提升配准质量，研究学者从特征检测、特征匹配及投影变换模型三个方面开展研究。

目前，已有很多研究者致力应用于图像拼接的特征检测算法，Lowe 等人[4]提出的 SIFT 特征具有尺度和旋转等不变性，环境适应性较好，由于采用特征点邻域内的梯度方向直方图，构造出高维浮点 SIFT 描述符，导致巨大的内存开销和匹配计算量[5]。为了提升检测效率，Ke 等人[6]提出 PCA-SIFT 算法，利用 PCA 降维，减少 SIFT 算法耗时，但对尺度变化和图像模糊较为敏感，鲁棒性仍有待提高。Calonder 等人[7]提出的 BRIEF 特征描述算法，利用二进制序列描述特征点，计算速度快，但该描述符不具备旋转和尺度不变性，拼接图像质量差。而 Bay 等人[8]提出的 SURF 特征充分利用积分图像特性，大大提高了描述效率，但检测到的特征点数量仅有 SIFT 算法的一半，影响了拼接质量。

特征匹配直接影响拼接算法的快速性和拼接误差，其核心是如何快速、有效地剔除误匹配对。Lin 等人[9]结合穷举搜索和

基金项目：国家自然科学基金资助项目（61673306）；江苏省重点研发计划资助项目（BE2016155）

作者简介：杨明东（1994-），男（白族），云南大理人，硕士研究生，主要研究方向为图像处理与模式识别；石英（1975-），女（通信作者），湖北武汉人，教授，博士，主要研究方向为图像处理、模式识别及车联网（a_laly@163.com）；华逸伦（1994-），男，硕士研究生，主要研究方向为图像处理与车联网；朱剑怀（1994-），男，硕士研究生，主要研究图像处理与模式识别．

最近邻比次近邻匹配策略进行特征匹配，但穷举搜索非常耗时，且最近邻比次近邻策略对阈值敏感，不合适的阈值会损失一部分正确的匹配对，造成匹配率低下。Hui 等人[10]使用 NCC 和 RANSAC 算法[11]进行特征匹配，NCC 需归一化互相关矩阵，计算耗时，且匹配精度低，使得后续的 RANSAC 算法收敛速度慢。韩天庆等[12]针对无人机航拍图像利用空间约束及双向匹配策略提高匹配精度，效果虽好，但其应用范围有限。

完成配准后，利用配准参数合成图像，为了降低拼接图像的扭曲变形，需要选择合适的投影变换模型。Brown 等人[13]利用全局捆绑调整完成旋转参数和相机焦距的估计进而完成多幅图像的投影，精度较高，但估计时间较长，不适用于快速拼接领域。Song 等人[14]以序列图像中的第一幅图像为起始参考图像，按帧到图像逐步扩大的策略将所有图像依次投影到参考平面，但每次拼接的误差都会累计到下一次的拼接，如果序列图像集合很大，将会造成严重的扭曲变形。Zaragoza 等人[15]提出 APAP 投影模型，可以精确地对齐图像，产生精度较高的拼接图像，但仍存在形状、面积失真问题。

针对上述问题，本文提出了快速且累计误差小的序列图像拼接算法，结合 SURF 描述符和 SIFT 特征的优势，并采用基于重合区域的特征检测算法，以提高检测效率；为了提升匹配精度，融合多种匹配策略，并结合坐标约束及 RANSAC 算法完成特征匹配；为了降低多次投影带来的累计误差，提出了利用配准参数计算任意图像到参考图像坐标空间的投影变换模型的方法；为了加快拼接速度，运用多线程技术合成全景图。最后设计了多组序列图像拼接测试，验证算法的有效性。

1　快速图像拼接算法框架

本文拼接算法框架设计如图 1 所示，分为预处理、图像配准及合成图像三个部分。预处理阶段根据全景图尺度估计将图像分辨率统一为 300×400，并标定重合区域。图像配准为经过特征检测和匹配后，估计图像间的变换矩阵得到配准参数。合成图像包括亮度校正、图像投影和图像融合等，最终实现快速且误差小的序列图像拼接。

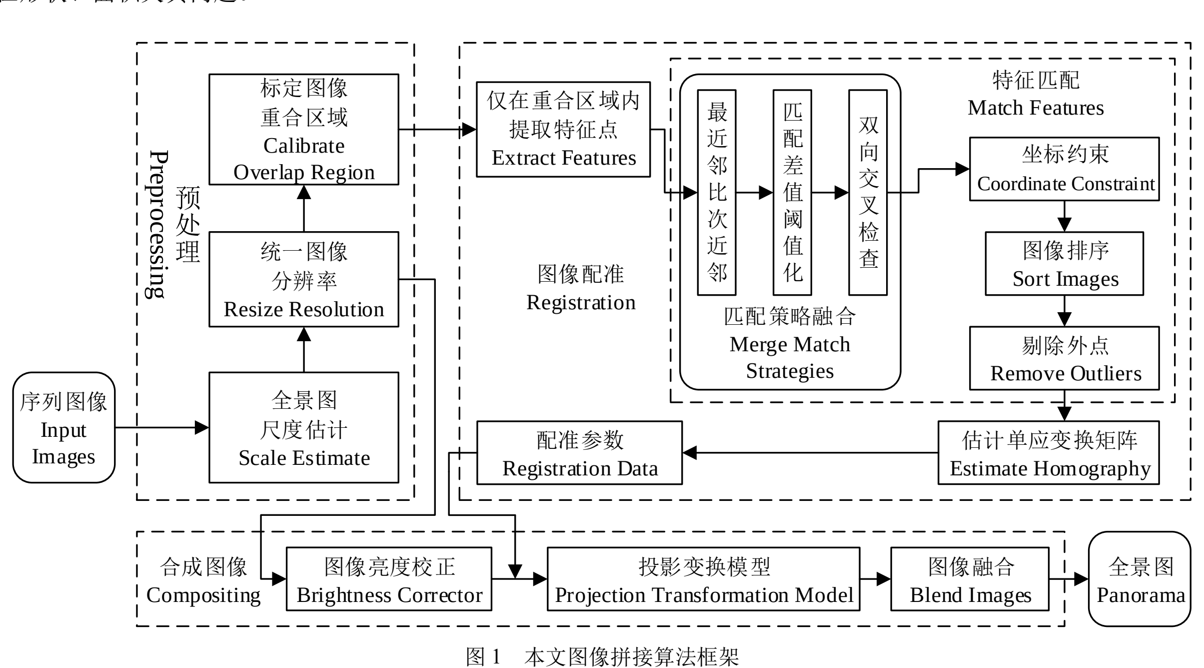


图 1　本文图像拼接算法框架

2　快速图像拼接算法原理

2.1　基于重合区域的快速特征检测算法

通常序列图像间的重合区域约为 10%~30%，本算法在该重合区域内提取特征点，较之整幅图像提取，减少了不必要的耗时。首先，标定重合区域，即标注每一幅图像的左右各 1/3 区域为检测区域。然后，提取该区域的 SIFT 特征点，再计算 SURF 特征描述符。本文特征检测算法具体步骤如下：

a)计算图像 $i(x,y)$的积分图像 $I_{sum}(m,n)$为

$$I_{sum}(m,n)=\sum_{x=0}^{x\le m}\sum_{y=0}^{y\le n}i(x,y) \tag{1}$$

b)构建尺度空间，即高斯金字塔 $L(x,y,\delta)$和高斯差分金字塔 $D(x,y,\delta)$：

$$L(x,y,\sigma)=G(x,y,\sigma)\otimes i(x,y) \tag{2}$$

$$D(x,y,\sigma)=(G(x,y,k\sigma)-G(x,y,\sigma))\otimes i(x,y) \tag{3}$$

其中：$G(x,y,\delta)$为变尺度的高斯核函数 $\frac{1}{2\pi\sigma^2}e^{-(x^2+y^2)/2\sigma^2}$；

c)在高斯差分金字塔 $D(x,y,\delta)$内，进行三维邻域内特征点极值搜索，初步确定特征点；

d)利用像元插值法定位亚像素级精度的特征点；

e)剔除插值偏移量大于 0.5、响应值小于 0.03 的点以及存在边缘效应的点；

f)通过积分图像 $I_{sum}(m,n)$，计算以特征点 *keypoint* 为中心的扇形区域内 *x*、*y* 方向的 *Haar* 小波响应值 *dx* 和 *dy*，统计幅值 $\sum dx+\sum dy$ 和对应的方向角 $\arctan(\sum dx/\sum dy)$，幅值最大的区域方向即为特征点主方向，且定义幅值大于最大幅值 80%的区

域方向为辅方向；

g)沿特征点主方向，取 20δ×20δ 的矩形区域，并将该区域划分为 4×4 个子块，然后统计积分图像 $I_{sum}(m,n)$子块的 *Harr* 响应值 *v*：

$$v=[\sum dx,\sum|dx|,\sum dy,\sum|dy|] \tag{4}$$

即可得到 4×4×4=64 维的特征描述符 *descriptor*，如果区分正负号，则为 128 维；

h)保存当前图像的特征点信息 keypoint 和 descriptor。

2.2　匹配策略融合的改进特征匹配算法

提取出特征点后，需要匹配物理位置一致的点，然后利用高精度的匹配对，计算图像间的单应变换矩阵。

2.2.1 常用匹配算法及其性能分析

常用匹配算法利用穷举搜索策略和欧氏距离度量准则，并结合最近邻比次近邻策略剔除误匹配对[9]，即计算图像 i_i 中的每个特征描述符 Des_{ik} 与图像 i_j 中所有特征描述符 Des_{jk} 间的距离 *Dis*：

$$Dis=\sqrt{\sum_{k=1}^{64}(Des_{ik}-Des_{jk})^2} \tag{5}$$

若最近距离与次近距离之比小于某一阈值，则认为匹配正确。该算法存在两个问题:a)若特征点数量庞大，需要遍历所有的 64 维特征描述符，效率低;b)最近邻比次近邻策略直接使用次近邻点计算次近邻距离，当最近邻 *d*(1)与次近邻 *d*(2)相差较小，而与其他近邻相差较大时，该匹配将被排除，这样会损失原本正确的匹配对，降低了正确匹配率。此外，仅使用一种策略仍会有较多的误匹配对，使得后续的精匹配算法效率降低。

2.2.2 改进的匹配算法

为了提升匹配精度，首先，利用距离平均法改进最近邻比次近邻策略，即用 *d*(2)~*d*(9)的平均值代替次近邻 *d*(2)，*d*(1)仍为最近距离，有效地避免了 *d*(1)与 *d*(2)相差较小时正确匹配对被误排除。

其次，融合匹配差值的阈值化和双向交叉检查策略，前者计算匹配对的特征描述符间的差值，若小于给定阈值，则认为匹配正确；后者需在两个方向上均搜索到相同的匹配对，即两个匹配点互为对方的最近邻点，则匹配正确。

最后，考虑到序列图像均处于同一方向，正确匹配对的横纵坐标有一定的关联，因此可增加坐标约束条件。以水平方向为例，(x_1,y_1) 和 (x_2,y_2) 是一对待约束的匹配点，令水平方向约束条件为

$$|y_1-y_2|<p_1,\ |w-x_1-x_2|<p_2 \tag{6}$$

其中:p_1，p_2 是两个阈值，本文分别取 p_1=10，p_2=8，图像的宽度 $w=300$，坐标约束原理示意如图 2 所示，其中 A、B 是正确的匹配对，C、D 由于不满足坐标约束条件将被排除。同理，垂直方向约束条件可以设为

$$|x_1-x_2|<p_1,\ |h-y_1-y_2|<p_2 \tag{7}$$

其中图像高度 *h*=400。

经过以上匹配后利用匹配数越多，图像越接近的思想进行图像排序，得到相邻图像间的匹配对集合。该集合中仍可能存在一两个误匹配点，本文采用 RANSAC 算法进行匹配对提纯，得到无误匹配的集合，对 *k* 组匹配对 (x_i,y_i)，(x_i',y_i')，$i\in 1,2,\cdots,k$，利用最小二乘法估计图像间的单应变换矩阵 *vectorH*，得到配准参数 $h=[h_0,h_1,\ldots,h_7]$：

$$h=(A^TA)^{-1}A^Tb \tag{8}$$

其中，$A=\begin{bmatrix} x_1' & y_1' & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1' & y_1' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_k' & y_k' & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_k' & y_k' & 1 \end{bmatrix}$，$b=\begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ \vdots \\ x_k \\ y_k \end{bmatrix}$。

为了提高匹配效率，Beis 等人[16]增加了优先级队列和超时机制，提出了基于 BBF 查询机制的 Kd 树搜索算法，其时间复杂度为 $O(nlogn)$，较之穷举搜索复杂度 $O(n^2)$，大大提高了高维大数据量的检索效率，因此，本文利用该搜索算法完成大量特征点的匹配工作。

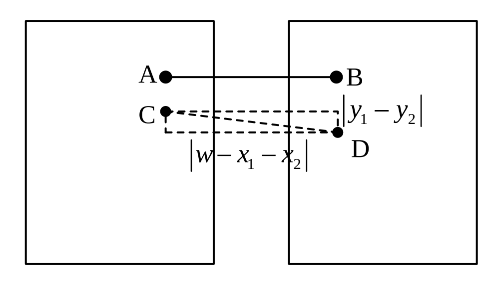


图 2　坐标约束原理示意图

综上，改进的特征匹配算法步骤为：

a)利用图像 i_i 和图像 i_j 的特征描述符建立 Kd 树，即 *kd_rootA* 和 *kd_rootB*；

b)利用 BBF 机制，在 *kd_rootA* 中查询 i_j 特征描述符的 9 个近邻点 *NeighborPoints*={N_1，N_2，…，N_9}，并计算与查询点的距离 *NeighborDis*={dis_1，dis_2，…，dis_9}；

c)求 N_2，N_3，…，N_9 这 8 个近邻点的平均距离，作为次近邻距离 *d*(2)，然后计算 $dis_1/d(2)$，若小于 0.65，则匹配成功，否则排除匹配对，得到匹配集合 *matches1*；

d)计算 *matches1* 中匹配对的特征描述符间的差值，若大于阈值 *maxFeatureDist*=0.4，则剔除该匹配项，得到匹配集合 *matches2*；

e)遍历图像 i_j 中所有特征描述符重复步骤 b)~d)，得图像 i_j 到图像 i_i 的匹配对集合 *goodmatches1*；

f)对图像 i_i 中所有特征描述符,重复 b)~e)四个步骤，得图像 i_i 到图像 i_j 的匹配对集合 *goodmatches2*；

g) 利用双向交叉检查策略，对 *goodmatches1* 和 *goodmatches2* 中的匹配对进行双向交叉检查，只有两个方向都能匹配到相同的匹配对，才是正确的匹配对 *matches3*；

h)对 *matches3* 中的每对匹配对 (x_1,y_1) 和 (x_2,y_2)，利用式（6）（7）的坐标约束条件进行约束，以得到高精度的匹配对集合 *matches4*；

i)利用 RANSAC 算法，剔除集合 *matches4* 中极少量的误匹配对，得到精确的匹配集合 *bestmatches*。

2.3 无缝序列图像并行拼接算法

利用单应变换矩阵进行图像投影，考虑到图像序列的光照、视角等亮度差异将影响拼接图像的视觉效果，需先进行亮度校正，分别求出重合区域的 R、G、B 三通道和，利用三通道间的比例系数进行亮度校正[17]。

传统序列图像的拼接算法主要使用帧到图像的逐帧扩大式拼接策略[14]，该策略缺陷有：a)直接串行拼接所有图像，必然消耗大量时间，影响拼接效率;b)每一帧拼接会累加上次的拼接误差，极易导致拼接失败。

针对上述问题，本文提出利用配准参数计算序列图像到参考图像坐标空间的投影变换模型的方法，首先假设参考图像索引值为 *m*，利用矩阵间的传递性，如图 3 所示，计算任意一幅图像到中间参考图像的投影变换矩阵 $toRefHvector_{im}$：

$$toRefHvector_{im}=\begin{cases}\prod_{i++}^{i=m-1} vectorH[i], & 当 i<m \\ \prod_{--i}^{i=m} vectorH^{-1}[i], & 当 i>m\end{cases} \tag{9}$$

其中：$i\in k,\ k=0,1,2,\cdots,n$，$vectorH[i]$表示相邻图像间的变换矩阵。然后以 $toRefHvector_{im}$ 作为投影变换模型，实现图像投影。假设参考图像的点集为 $(x_i,y_i,1)$，其他任意图像的点集为 $(x_i',y_i',1)$，则有

$$\begin{bmatrix}x_i\\y_i\\1\end{bmatrix}=\begin{bmatrix}h_0^i & h_1^i & h_2^i\\h_3^i & h_4^i & h_5^i\\h_6^i & h_7^i & 1\end{bmatrix}*\begin{bmatrix}x_i'\\y_i'\\1\end{bmatrix} \tag{10}$$

其中，h_0^i、h_1^i、h_2^i、h_3^i、h_4^i、h_5^i、h_6^i、h_7^i 是第 i 幅图像到参考图像的变换矩阵 $toRefHvector_{im}$ 参数，可依次将所有图像的点集利用式（10）投影到参考坐标平面。

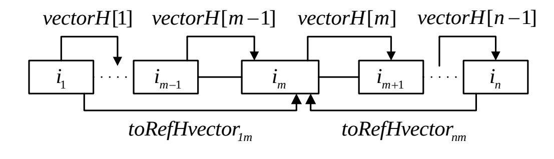


图 3　变换矩阵关系图

该投影变换模型没有携带上一次的拼接误差，每次的投影误差只与原始图像间的配准误差有关，扭曲变形小，并且以序列图像中间位置为参考，拼接误差仅为中间图像到两端图像的误差累计，相比传统拼接策略误差几乎减少了一半，累计误差可近似表达为

$$E_1=\max(\sum_{i=1}^{m}e_i,\sum_{j=m}^{n}e_j) \tag{11}$$

传统拼接策略的累计误差则为

$$E_2=\sum_{i=1}^{n}e_i \tag{12}$$

其中：e 表示相邻图像间的配准误差，n 为序列图像的数量，当 n=2 时，两种策略误差近似相等，随着 n 的增大，E_1 与 E_2 差异越大，误差降低越明显。

此外，如果将所有图像顺序投影，输入图像数量 n 很大时，仍会有较大耗时，因此本文利用多线程技术，提升图像拼接效率，以参考图像为界，将序列图像分为 $LiftImages=\{i_1,i_2,\cdots,i_m\}$ 和 $RightImages=\{i_m,i_{m+1},\cdots,i_n\}$ 两部分，其拼接过程各采用一个线程，分别将 *LiftImages* 和 *RightImages* 中的每一幅图像投影到参考平面，仅当两个线程工作完成时，才合成它们的结果，并采用渐入渐出的加权平均融合算法消除拼接缝，提高全景图像的质量。

3　实验分析与比较

本文实验均使用硬件配置为 i5-3230M CPU、2.6 GHz 主频及 8 GB 内存的计算机，系统为 Windows7 64 位。使用 Qt Creater5.7 开发平台并结合 OpenCV 2.4.10 视觉库和 boost 多线程库实现本文图像拼接算法。

3.1　实验设置及数据集

为了验证本文拼接算法的有效性，针对快速性设置了特征检测与匹配耗时实验，而对于误差设置了检测鲁棒性、匹配准确性及拼接累计误差实验。

实验使用 Mikolajczyk 数据集[18]中的 4 组图像序列，包括光照变化数据集 leuven、尺度和旋转变化数据集 boat、图像模糊数据集 bikes 和视角变化数据集 graf，每组包括相同场景下不同变化程度的 6 幅图像，该数据集包括不同环境下的干扰，利于比较检测算法鲁棒性和匹配性能。

3.2　特征检测算法快速性及鲁棒性比较

3.2.1 快速性比较

取 4 组数据集中的第 1 幅图像，分别利用本文特征检测算法和传统 SIFT 算法对其进行特征检测，均使用算法默认参数，表 1 统计了提取、描述及总耗时。

结果表明，本文检测算法和 SIFT 算法均检测到了相同数量的特征点，且提取特征点时间基本相同，这是由于本文特征提取与 SIFT 算法保持一致。而本文特征描述时间比 SIFT 算法少 0.41~2.03s 左右，因为本文算法没有使用维数高、计算量大的 SIFT 描述符，而是利用积分图像的特性快速地计算 64 维 SURF 描述符。因此，本文特征检测算法的检测效率优于传统 SIFT 算法，应用于图像拼接优势明显。

3.2.2 鲁棒性比较

利用匹配分数对比本文检测算法与 SIFT 算法的鲁棒性，匹配分数定义为图像间匹配对的数量与两幅图像中检测到的较少的特征点数量之比。利用包括不同干扰的 4 组数据集按照欧式距离最小的原则进行匹配。图 4 统计了本文算法和 SIFT 算法在 4 种干扰下的匹配分数。匹配分数越高，鲁棒性越好。

结果表明，在视角变化、尺度缩放、光照变化及模糊 4 种干扰下，本文算法使用的 SURF 描述符在鲁棒性上略优于 SIFT 描述符，因此，本文算法在保证快速性的前提下，同时也保证了算法的鲁棒性能。

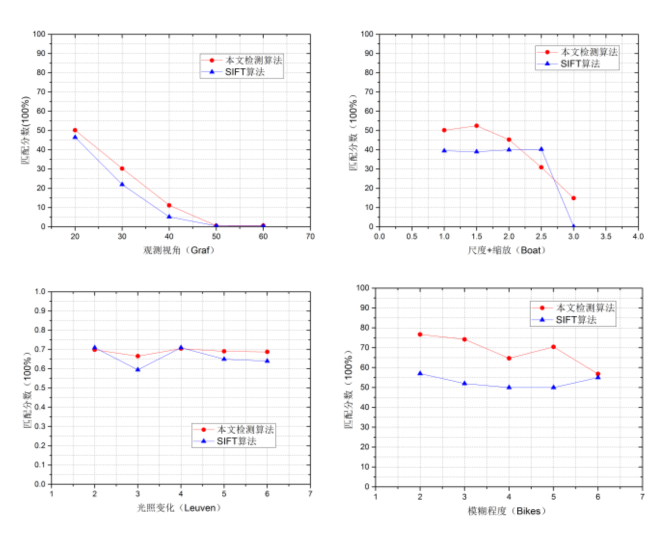


图 4　检测算法鲁棒性比较

3.3　特征匹配算法快速性与准确性比较

3.3.1 快速性比较

利用本文匹配算法和 Lin 匹配算法[9]对 4 组数据集中前两张图像进行特征匹配，图 5 给出了匹配时长对比结果。

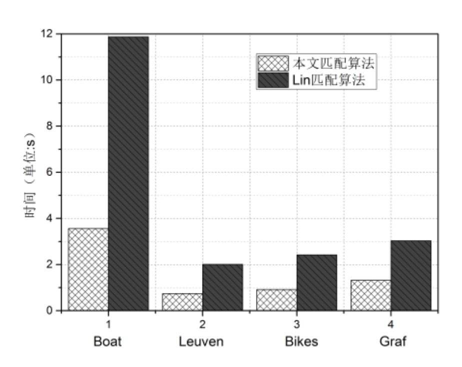


图 5　匹配算法快速性比较

数据集中的图像分辨率均在 600×800 以上，特征点数量大，传统的匹配算法需要全部遍历，匹配十分耗时，尤其 boat 数据集提取了 8547 个特征点，匹配需 12s 左右，显然无法胜任快速拼接任务。而本文匹配算法使用了改进的 Kd 树搜素算法，其计算复杂度随着特征点数量的增多而降低，使得匹配效率提高了 2.30~3.33 倍。

表 1　特征检测算法快速性比较

数据集	SIFT 算法				本文算法			
	特征点数量	提取时间（s）			特征点数量	提取时间（s）		
		提取	描述	总		提取	描述	总
boat	8547	2.19	1.18	3.37	8547	2.09	0.77	2.86
leuven	2117	1.17	1.74	2.91	2117	1.15	0.22	1.37
bikes	1728	1.41	1.81	3.22	1728	1.38	0.21	1.59
graf	3061	1.24	2.32	3.56	3061	1.23	0.29	1.52

3.3.2 准确性比较

准确性可以从正确匹配百分比和平均距离两个方面比较[19]，每组数据集都以第一幅图像为参考图像，然后分别与其他图像进行特征匹配。图 6 显示了本文匹配算法、Lin 匹配算法及 Hui 匹配算法[10]在不同环境下的不同变化程度的正确匹配百分比与平均距离的对比结果。

其中，正确匹配百分比 R_r 为：

$$R_r = \frac{N_{ransac}}{N_r} \times 100\% \tag{13}$$

其中，N_{ransac} 为正确匹配对数，N_r 为两幅待拼接图像初始匹配对数。而平均距离是指匹配过程中，所有正确匹配对描述符间的平均距离 *disAvg*：

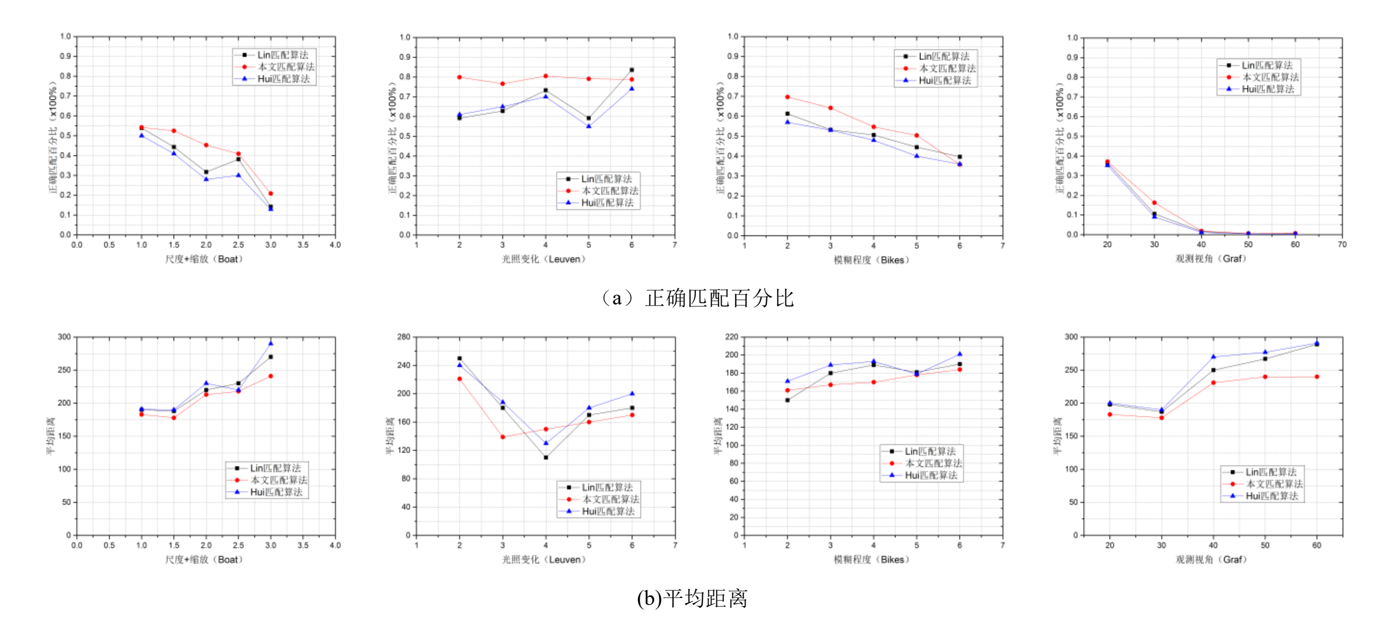
$$disAvg = \frac{1}{N_{ransac}} \sum_{i=1}^{N_{ransac}} dis_i \tag{14}$$

其中，dis_i 表示第 i 对正确匹配对描述符间的欧式距离，*disAvg* 间接表达了匹配的准确性。

图 6（a）显示了正确匹配百分比的实验结果。对于尺度和缩放变化，3 种匹配算法在尺度变化很大时，正确匹配百分比迅速降低，但本文算法均比其他两种算法高 10%左右，准确性较高；对于光照变化，3 种算法均能保持 55%以上的正确匹配百分比，但本文匹配算法实验曲线更平稳，基本能保持在 80%左右，而 Lin 算法和 Hui 算法随着光照强度的不同，正确匹配百分比波动较大，鲁棒性较差；对于图像模糊，当模糊因子较小时，本文匹配算法能保持 70%左右的正确匹配百分比，随着模糊因子的增加，匹配准确性有所降低，但其准确性仍优于其他 2 种算法；对于视角变化，3 种算法在一定范围内能保持一定的正确匹配百分比，但变化范围过大时，3 种算法均不能保证视角不变性，准确性较差，由于视角变化较大时 SURF 描述符的鲁棒性降低，同时视角变换使得图像不在同一方向因而不再满足约束条件，使得正确匹配率大大降低，但本文算法仍略好于其他 2 种算法。

图 6（b）为平均距离实验结果。对于尺度和旋转变化，随着尺度因子的增加，3 种算法的平均距离都在增加，说明匹配准确性降低，但本文匹配算法的平均距离比 Lin 算法低 20 左右，比 Hui 算法低 30 左右；对于光照变化，当光照程度为 3.5 倍时，两种算法的平均距离减少较为明显，说明该亮度匹配准确性较高，本文匹配算法的稳定性优于其他 2 种算法；对于模糊变化，本文算法得到的曲线很平稳，均能保持在 170 左右，而 Lin 算法曲线波动较为明显，匹配性能差，Hui 算法在模糊因子为 6 时，平均距离大幅度增加；对于视角变化，当视角小

范围变化时，3 种算法平均距离较低，在 185 左右，而当视角继续增加至 40 度以上时，平均距离迅速增加，匹配准确性大大降低，但总体看来，本文匹配算法平均距离要低于其他 2 种算法。



（a）正确匹配百分比

(b)平均距离

图 6　匹配算法准确性比较

综上，本文匹配算法将多种匹配策略融合，并使用坐标约束及 RANSAC 算法，不仅有效地避免了原本正确匹配对被排除，而且大大增加了正确匹配率，准确性高。

3.4　拼接算法快速性与累计误差比较

3.4.1 快速性比较

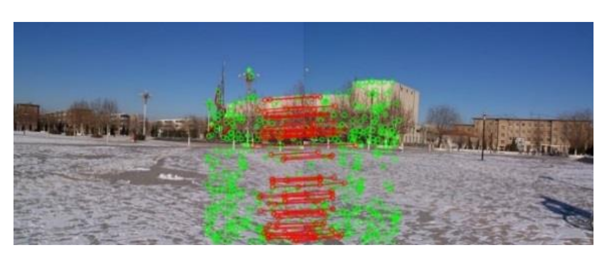
为了验证本文拼接算法的快速性，设置了两组实验：两幅图像和多幅图像的拼接，以研究图像数量对快速性的影响。其中实验 1 使用 Adobe Panoramas Dataset 图像集中的两幅图像（如图 7 所示）进行拼接测试，结果如图 8 所示，图 8（a）展示了本文特征检测算法对标定区域分别提取的 163、174 个特征点；图 8（b）显示了特征粗匹配结果， 24 对匹配成功，其中有一对明显的误匹配；图 8（c）给出了 RANSAC 算法提纯的 21 个匹配对，正确匹配百分比高；图 8（d）则给出了图像投影和加权融合后的拼接图，拼接变形小，视觉效果较好。实验 2 使用网上下载的 14 幅序列图像进行拼接测试（如图 9（a）所示），图 9（b）结果表明，图像内黑色像素比例明显高于两幅图像的拼接结果，也有一定的扭曲变形，但仍能保持较好的视觉效果。



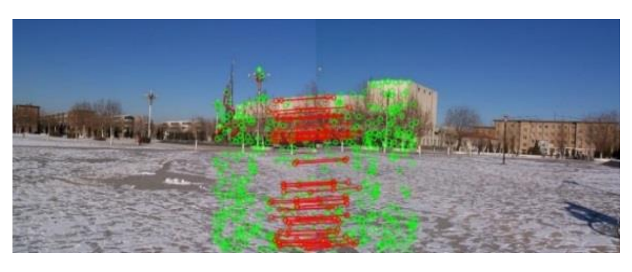

图 7　Adobe Panoramas Dataset 图像集




(a)特征检测



(b) 特征粗匹配（有明显地一对错误匹配对）



(c)特征精匹配



(d)拼接结果

图 8　2 幅图像拼接结果



（a）输入序列图像



（b）拼接结果

图 9　14 幅序列图像拼接结果

最后，统计实验 1 和实验 2 的拼接总耗时，并与拼接领域常用的 Brown 拼接算法[13]及 Song 拼接算法[14]进行比较，结果如表 2 所示。实验 1 本文算法比 Song 算法快 0.634s，而 Brown 算法耗时最多，为 2.685s。实验 2 中 Brown 拼接算法耗时长达 89.64s，明显高于 Song 和本文拼接算法，难以用于快速拼接领域，因为需要长时间的参数估计和多分辨率融合，时间复杂度高；Song 拼接算法耗时 9.114s，由于无须参数估计，且采用基于亮度的加权平均融合算法，相比 Brown 算法效率大大提升；本文拼接算法耗时仅 3.569s，比 Song 算法更优，此外，随图像数量的增加，本文拼接算法的快速性优势更加显著。

表 2　拼接总耗时比较

实验	总耗时(单位：s)		
	Brown 拼接算法	Song 拼接算法	本文拼接算法
实验 1	2.685	0.987	0.353
实验 2	89.640	9.114	3.569

本文拼接算法的运算复杂度优势主要体现在：（1）统计 SURF 特征描述符矢量时，使用积分图像统计一个 Haar 特征的复杂度为 $O(1)$，与遍历 Haar 模板内所有像素复杂度 $O(n^2)$相比，复杂度大大降低；（2）在特征匹配时使用改进的 Kd 树算法，与穷举匹配相比，时间复杂度从 $O(nlogn)$变为 $O(n^2)$，优势明显；（3）在拼接时，运用多线程技术，理论上可以将复杂度降低一倍；（4）使用了像素级的融合算法，复杂度较之多分辨率融合大大降低。因此，本文算法快速性优势明显。

3.4.2 累计误差比较

目前，图像拼接领域尚未有公开、完整的误差评价标准，本文将使用扭曲度 *Twist* 和有用像素比例 *Correctness* 评价拼接误差[20]，其定义分别为：

$$Twist = Max\{\ Slopes(i,j)\ \}\ i,j \in ResultPoint \tag{15}$$

$$Correctness = 1 - \frac{BlackPixels}{TotalPixels} \tag{16}$$

其中，*ResultPoint* 为图像变换后的中心坐标集合，*Slopes*(*i*,*j*)为相邻坐标间的斜率。*BlackPixels* 为拼接图中的黑色无用像素点的个数，*TotalPixels* 为总的像素个数。扭曲度越低，有用像素比例越高，表示累计误差越小。

利用本文拼接算法及 Song 拼接算法，分别拼接图 9(a)所示的 4~14 幅图像，表 3 列出了每组实验的结果。结果表明，随着拼接图像数量的增加，扭曲度逐渐增加，有用像素逐渐减少，累积误差有小幅增长。但本文拼接算法得到的扭曲度基本保持在 0.35 以下，有用像素比例保持在 95%以上，相比 Song 拼接算法扭曲度低 0.07~0.13，有用像素比例则增加 5.62%~13.31%，且图像数量越多，本文拼接算法累计误差增长幅度越低，优势越明显。尤其是对 8 幅以下的拼接任务，扭曲度低于 0.20，有用像素比例均高于 96%，累计误差低，有较高的实用价值。

表 3　扭曲度、有用像素比例比较

实验	Song 等拼接算法		本文拼接算法	
	Twist	Correctness	Twist	Correctness
4 幅	0.1301	91.11%	0.0631	96.73%
6 幅	0.2189	90.06%	0.1357	96.59%
8 幅	0.2822	89.88%	0.1995	96.23%
10 幅	0.3534	87.05%	0.2433	96.14%
12 幅	0.4244	83.78%	0.3156	95.04%
14 幅	0.4676	82.62%	0.3328	95.93%

4　结束语

本文为了提高匹配准确率、降低拼接时间及累计误差，提出了基于多匹配策略融合的改进拼接算法。该算法首先采用基于重合区域及 SURF 描述符的 SIFT 特征检测算法，以提高特

征检测速度。其次，融合多种匹配策略，并结合坐标约束，在光照变化、尺度和旋转变化、图像模糊、视角变化等环境下，特征匹配准确率更高。最后，提出序列图像到参考图像坐标空间的投影变换模型的计算方法，以降低多次拼接带来的累计误差，并利用多线程技术提升拼接速度，最终得到高质量的全景图。但本文使用了加权融合算法，当序列图像中存在运动物体时，会出现鬼影、拼接缝等，在未来的工作中将继续研究既能解决以上问题，又能快速完成融合的算法。

参考文献：

[1] Wang Guodong, Zhai Zhengjun, Xu Bangdao, et al. A parallel method for aerial image stitching using ORB feature points [C]// Proc of the 16th IEEE International Conference on Computer and Information Science. 2017: 769-773.

[2] Li Yuelong, Vishal Monga. SIASM: sparsity-based image alignment and stitching method for robust image mosaicking [C]// Proc of IEEE International Conference on Image Processing. 2016: 1828-1832.

[3] 任刚, 彭冬亮, 谷雨. 基于圆柱面映射的快速图像拼接算法 [J]. 计算机应用研究. 2017, 34 (11): 1-8.

[4] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints [J]. International Journal of Computer Vision, 2004, 60 (2): 91-110.

[5] Miksik O, Mikolajczyk K. Evaluation of local detectors and descriptors for fast feature matching [C]// Proc of the 21st International Conference on Pattern Recognition. 2012: 2681-2684.

[6] Ke Y, Sukthankar R. PCA-SIFT: a more distinctive representation for local image descriptors [C]// Proc of IEEE Computer Society Conference on Computer Visi-on and Pattern Recognition. 2004: 511-517.

[7] Calonder M, Lepetit V, Strecha C, et al. BRIEF: binary robust independent elementary features [C]// Proc of the 11th European Conference on Computer Vision. 2010: 778-792.

[8] Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF) [J]. Computer Vision & Image Understanding. 2008, 110 (3): 346-359.

[9] Lin Mingxin, Xu Gang, Ren Xingning, et al. Cylindrical panoramic image stitching method based on multi-cameras [C]// Proc of IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems. 2015: 1091-1096.

[10] Hui Qi, Ma Yajie, Yi Hu, et al. A novel feature matching algorithm against moving background for indoor video surveillance [C]// Proc of the 36th Chinese Control Conference. 2017: 11510-11514.

[11] Fischler M, Bolles R. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography [J]. Communications of the ACM, 1981, 24 (6): 381-395.

[12] 韩天庆, 赵银娣, 刘善磊, 等. 空间约束的无人机影像 SURF 特征点匹配 [J]. 中国图象图形学报. 2013, 18 (06): 669-676.

[13] Brown M, Lowe D G. Automatic Panoramic Image Stitching using Invariant Features [J]. International Journal of Computer Vision, 2007, 74 (1): 59-73.

[14] Song F, Lu B. An automatic video image mosaic algorithm based on sift feature matching [C]// Proc of International Conference on Communication, Electronics and Automation Engineering. 2013: 879-886.

[15] Zaragoza J, Chin T J, Brown M S, et al. As-projective-as-possible image stitching with moving DLT [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2339-2346.

[16] Beis J S, Lowe D G. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 1997: 1000-1006.

[17] Xiong Y, Pulli K. Color correction for mobile panorama imaging [C]// Proc of the 1st International Conference on Internet Multimedia Computing and Service. 2009: 219-226.

[18] Mikolajczyk K, Schmid C. A performance evaluation of local descriptors [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2005, 27 (10): 1615-1630.

[19] 张莹, 闫璠, 高赢, 等. 基于 ORB 算法和 OECF 模型的快速图像拼接研究 [J]. 计算机工程与应用, 2017, 53 (1): 183-189.

[20] 瞿中, 林嗣鹏, 鞠芳蓉. 一种改进的降低扭曲误差的快速图像拼接算法 [J]. 计算机科学, 2016, 43 (5): 279-282.